Developing A Web-based Workbook for English Supporting the Interaction of Students and Teachers

Björn Rudzewitz Ramon Ziai Kordula De Kuthy Detmar Meurers
Seminar für Sprachwissenschaft, SFB 833, LEAD
Universität Tübingen
{brzdwtz,rziai,kdk,dm}@sfs.uni-tuebingen.de

Abstract

We discuss the development of FeedBook, an intelligent electronic workbook supporting the teaching of English as Foreign Language in German secondary school. The overarching goal is to address real-life formal education needs using current Natural Language Processing technology.

Our interactive, web-based workbook is based on the print workbook of a schoolbook officially approved for 7th grade English classes in secondary schools in Baden-Württemberg, Germany. The workbook offers a range of activities for students that typically are assigned by instructors as homework to accompany the regular English classes. In our web-based version, students can complete activities online, submit them to their teacher, and view teacher feedback. Teachers are supported in providing both formative and summative feedback to individual students by an auto-correct and feedback memory system, and they can view aggregates of student performance.

This article describes the development of the web-based workbook and its use from a technical and pedagogical perspective. We are currently working on adding automatic immediate feedback to learners that is designed to incrementally support individual learners in successfully completing a given task.

1 Introduction

Research in Second Language Acquisition (SLA) and Foreign Language Teaching (FLT) has stressed the importance of individualized, immediate feedback on learner production for learner proficiency development (e.g., Mackey, 2006). In the classroom, the teacher is generally the only source of reliable, accurate feedback available to students, which poses a well-known practical problem: in a class of 30 students, despite individual differences warranting individual feedback to students, it is highly challenging for a teacher to provide it in class or, in a timely fashion, on homework.

One of the means to address this problem are Intelligent Tutoring Systems (ITS), which have been advocated since the 90's. Such systems are successfully used in domains such as mathematics (Sabo et al., 2013), where the system input can be sufficiently constrained to support automatic evaluation. In the language domain, ITS are much less common, although there has been extensive related research (cf., Heift and Schulze, 2007).

In order to address the gap between research on intelligent language tutors, foreign language teaching insights, and real-life classroom usage, we are developing FeedBook, a web-based English workbook we are creating in collaboration with Diesterweg of the Westermann Gruppe, a major German schoolbook publisher.[1] The ultimate goal of the system is to provide individualized and immediate scaffolding feedback to learners in order to guide them towards solutions for a number of different activities. In the current stage, the system provides a web-based implementation of the traditional print workbook which enables students to complete activities and teachers to give formative and summative feedback on the language produced by the student. The teacher is assisted through several automatic mechanisms, ranging from the system remembering previously given feedback to annotation suggestions based on Natural Language Processing (NLP) technology.

[1] https://verlage.westermanngruppe.de

Björn Rudzewitz, Ramon Ziai, Kordula De Kuthy and Detmar Meurers 2017. Developing a web-based workbook for English supporting the interaction of students and teachers. *Proceedings of the Joint 6th Workshop on NLP for Computer Assisted Language Learning and 2nd Workshop on NLP for Research on Language Acquisition at NoDaLiDa 2017*. Linköping Electronic Conference Proceedings 134: 36–46.

The paper is organized as follows: Section 2 briefly discusses some existing language tutoring approaches. In section 3, we discuss the challenges involved in creating a web-based workbook based on a traditional print workbook. Section 4 then discusses FeedBook and its different components in detail, before section 5 concludes the paper with an outlook on next steps of the project.

2 Related Work

Intelligent Language Tutoring Systems (ILTS) proposed in the literature range from highly ambitious conversation machines (e.g., DeSmedt, 1995) to more modest workbook-like approaches (e.g., Heift and Nicholson, 2001; Nagata, 2002). However, as discussed by Heift and Schulze (2007), most of the systems are research prototypes that have never seen real-life testing or use. Following Amaral and Meurers (2011), we describe three notable exceptions below.

Robo-Sensei (Nagata, 2009) is a system for Japanese which presents a series of exercises for each of its 24 lessons. The activities are contextualized, including visual aids and picture material on Japan. The system was created by Noriko Nagata, who also designed the Japanese teaching curriculum. The effectiveness of the system has been investigated explicitly (Nagata, 1993; 1996; 1997), confirming that *Robo-Sensei* supports the acquisition of grammar and vocabulary. *Robo-Sensei* generally constrains the language that the learner enters into the system by providing English cues, thus essentially presenting contextualized translation exercises.

The *E-Tutor* (Heift, 2003) is an ILTS for German developed by Trude Heift and integrated into the German curriculum of her department at Simon Fraser University. German students in the program complete *E-Tutor* exercises as regular class requirement, which has enabled Trude Heift to research the system's effectiveness and further its development (Heift, 2001; 2004; 2005). The system contains four types of exercises with NLP support: *i)* provide the missing word, *ii)* build sentences with the words given, *iii)* translate a phrase, and *iv)* write down the sentence read by the system. The exercise types explicitly constrain the learner productions through the words and phrases given in the prompt.

TAGARELA (Amaral and Meurers, 2011) is a tutor system for Portuguese. It is a web-based workbook featuring six activity types: listening comprehension, reading comprehension, rephrasing, description, fill-in-the-blanks and vocabulary practice. The system was integrated into the Portuguese program at The Ohio State University and the University of Massachusetts Amherst, where Luiz Amaral integrated it into individualized instruction, regular courses, and distance learning courses. Some of the activity types explicitly constrain the learner productions, whereas the comprehension tasks attempt to do so implicitly through the contents. To make the system more flexible in terms of activity types and user demands, a modular architecture based on UIMA (Ferrucci and Lally, 2004) was developed (Amaral et al., 2011), including a complete rewrite of the NLP components.

Overall, there have been only few attempts to connect state-of-the-art NLP and insights from SLA with actual widespread classroom usage, and the developers of the tutoring system were also involved in the actual teaching.

3 Adapting a Print Workbook for the Web

In contrast to the related work just discussed, the system introduced here takes as starting point an existing workbook that already is integrated into the real-life formal education context. To accomplish the goal of improving the learning experience for the students and support the teachers with minimal overhead, a crucial requirement for our development was to make the digital version as similar as possible to the look and feel to the print version. Users familiar with the print workbook thus can directly benefit from the added value of the web-based version without first requiring training.[2]

The web-based workbook is based on the print version of the schoolbook *Camden Town Gymnasium 3*, approved for 7th grade English classes in German secondary schools. As a first step moving from a print workbook to a web-based version, it is necessary to decide which activities to include and what a general data model representing them should look like. We therefore analyzed and categorized all activities in the print schoolbook in order to group similar exercises together and form a category system that generalizes over all group

[2] To be able to address any issues arising in real-life use, the system also includes the functionality to contact the developers with bugs or feature requests.

instances. To this end, we analyzed all activities of the print workbook in terms of:

(i) activity type, e.g., reading comprehension, fill-in-the-blank

(ii) expected well- or ill-formed language variation: How is the space of possible learner productions restricted by the instruction and material (cf. Quixal and Meurers, 2016; Meurers and Dickinson, 2017)

(iii) expected type of system input, e.g., check marks, single characters, words, a sentence, short texts

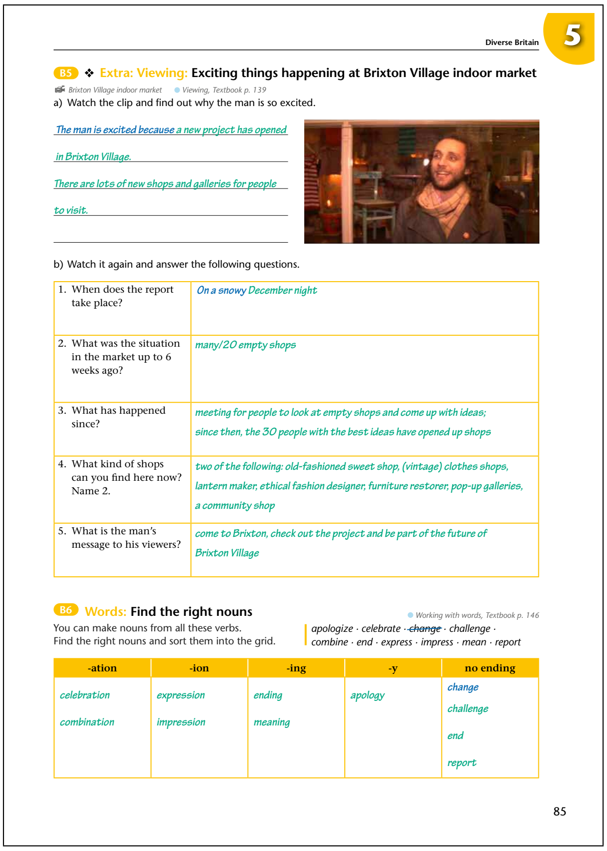
(iv) form or meaning-orientation

(v) language forms targeted explicitly or implicitly, e.g., simple past vs. past perfect.

Given our overall goal of linking real-life needs to NLP-based solutions, our primary focus is on activities for which the learners produce language (as opposed to, e.g., check marks), but where the language produced is constrained enough to determine potential reference answers (which rules out, e.g., activities asking about the students' background). We also aimed to cover both meaning- and form-oriented tasks, as both aspects are essential in FLT and contained in current workbooks.

Development first concentrated on two activity types: short answers (requiring one or more free-text sentences, usually in a meaning-oriented activity), and fill-in-the-blanks (a text with gaps, usually lexical content or form-oriented). Since some of the activities we chose also contain preparatory sub-tasks involving mapping (text or images that need to be matched) or variants of multiple-choice, we also included those task types to be able to offer the same activity sequences as in the print workbook. We currently cover 55 of the 177 workbook activities (some containing preparatory sub-tasks). To support a systematic use of the web-based workbook as a straightforward replacement of the print version in real-life teaching, we are currently adding further activity types. As also argued in Amaral and Meurers (2011), satisfying real-life education needs using an ILTS is as dependent on the user interface, web-programming, and related visual and task design issues as it is on the NLP analysis.

Figure 1 provides an example task from the *Camden Town 3* print workbook, where students are asked to watch a video clip provided on a DVD and answer questions.[3] In Figure 4, we will later see the web-based adaptation of the same task.


Diverse Britain

5

B5 ❖ Extra: Viewing: Exciting things happening at Brixton Village indoor market

Brixton Village indoor market ● Viewing, Textbook p. 139

a) Watch the clip and find out why the man is so excited.

The man is excited because a new project has opened in Brixton Village.

There are lots of new shops and galleries for people to visit.

b) Watch it again and answer the following questions.

1. When does the report take place?	On a snowy December night
2. What was the situation in the market up to 6 weeks ago?	many/20 empty shops
3. What has happened since?	meeting for people to look at empty shops and come up with ideas; since then, the 30 people with the best ideas have opened up shops
4. What kind of shops can you find here now? Name 2.	two of the following: old-fashioned sweet shop, (vintage) clothes shops, lantern maker, ethical fashion designer, furniture restorer, pop-up galleries, a community shop
5. What is the man's message to his viewers?	come to Brixton, check out the project and be part of the future of Brixton Village

B6 Words: Find the right nouns

● Working with words, Textbook p. 146

You can make nouns from all these verbs.
Find the right nouns and sort them into the grid.

apologize · celebrate · ~~change~~ · challenge · combine · end · express · impress · mean · report

-ation	-ion	-ing	-y	no ending
celebration combination	expression impression	ending meaning	apology	change challenge end report

85

Figure 1: Example task from the print workbook © *Westermann Gruppe*

Following common software engineering practice, in FeedBook we separate display from function and content. The display of an activity is determined by its activity type rather than being hard-coded for individual activities. This readily supports adding activities of the established types at a later point by providing only the new content. This implies that the data model for activities contains both textual content as well as links to context media such as graphics or audio. The activity data model also integrates all components required for the NLP analysis to provide feedback. For example, for processing short answers in terms of the semantic appropriateness, the feedback system needs direct access to the prompt text (reading question), the target answer, and the student answer (cf., e.g., Meurers et al., 2011b).

In the next section, we present the FeedBook functionality and how it integrates the activities and challenges characterized in this section.

[3] In this teacher version of the workbook, target answers are shown in green.

4 The FeedBook System

The FeedBook is designed as a multi-layer web application. The system is platform-independent, only requiring a computer, tablet, or smartphone with a web browser and internet access. The current version of the system supports a common workflow in German secondary schools: Students work on exercises, typically after being assigned those exercises as homework by the teacher, and submit their results to the teacher. The teacher corrects the submission and sends it back to the student with the feedback. The system provides automatic error annotation assistance and a feedback memory to facilitate the work of the teacher. The student can inspect the teacher feedback, and a diagnostics interface allows the teacher to identify general problems. Since the system is currently used by German students at a lower to intermediate level of English proficiency, German is used as interface and meta language, though this can be configured differently.

The purpose of the FeedBook is *not* to replace the class or the teachers, but to provide an opportunity for students to individually practice using online exercises at any time. It also relieves teachers from the repetitive work of providing feedback on the same issues over and over again, while at the same time allowing them to view aggregates of student performance so that they can use the class time efficiently to target common issues or misconceptions.

In the following, the system as developed in the first phase of the project is described in more detail, following the workflow just outlined. While for the second phase of the project, we are working on system components that provide immediate individual scaffolding feedback to students while they work on the activities, the teacher feedback data currently being collected is already used to reduce the work load of the teacher through components included in the discussion below.

4.1 The Lobby

The lobby is the central starting point loaded after a standard authentication process requiring a user name and password. The structure and functionality of the lobby differs by the role of the user (teacher, student, administrator) who logged in to accommodate their different needs.

Student Lobby For students, the lobby presents a table of contents hosting all exercises as tiles in a mosaic design grouped by the book themes. Figure 2 shows the lobby after selection of Theme 5 “Diverse Britain”.

After a student worked on an exercise, the exercise tile with the exercise title and page number[4] also indicates whether an exercise has been saved by the student, submitted to the teacher, or corrected by the teacher. In the latter case, a check mark using a traffic light color scheme indicates the overall teacher rating, and an information symbol indicates that the teacher provided annotations of errors or other comments.

When clicking on or touching an exercise tile, FeedBook forwards the student to the Practice Interface (section 4.2) or to the Result Interface showing feedback (section 4.4), depending on the state of the exercise.

Teacher Lobby Teachers have a choice between two conceptually and functionally distinct variants of the lobby: (a) tree view, (b) submission table.

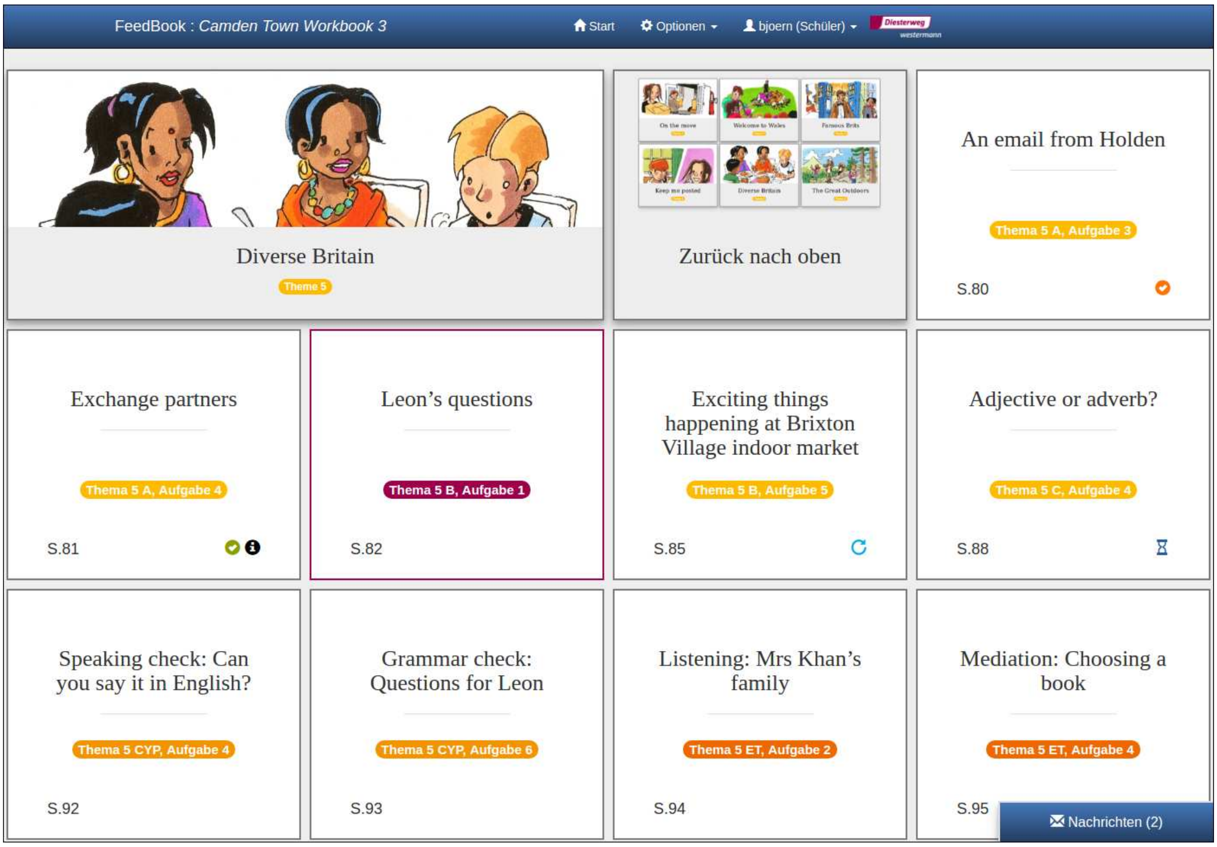
In variant (a), the teacher is shown the two column layout illustrated in Figure 3). In the left column, a table of contents of the workbook is shown, including only activities to which submissions of the teacher’s students exist. On selection of an activity, a submission table is shown in the right column listing all associated students (the class), the time they submitted the activity, their detailed submission status (correction pending, partially corrected, correction complete), the number of error annotations added to this submission, and the overall rating. Selecting a submission via click or touch forwards the teacher to the correction view (section 4.4).[5]

In variant (b), the teacher sees a list widget with all submissions. This includes all of the information mentioned above, but for all submitted exercises. The list can be filtered to only show submission that still require teacher interaction so that this view can serve as a To-Do list for teachers. This view also permits direct selection of submissions by click or touch.

Two buttons are shown at the table bottom. The first serves to automatically mark all submissions that are identical to the target answers provided by the teacher workbook as correct. The second button sends a reminder to all student who have not yet submitted the exercise.

[4] The page number is included for ease of cross-reference with the print workbook.

[5] Teachers can revisit and revise corrections incrementally.



Diverse Britain
Theme 5

Zurück nach oben

An email from Holden
Thema 5 A, Aufgabe 3
S.80

Exchange partners
Thema 5 A, Aufgabe 4
S.81

Leon’s questions
Thema 5 B, Aufgabe 1
S.82

Exciting things happening at Brixton Village indoor market
Thema 5 B, Aufgabe 5
S.85

Adjective or adverb?
Thema 5 C, Aufgabe 4
S.88

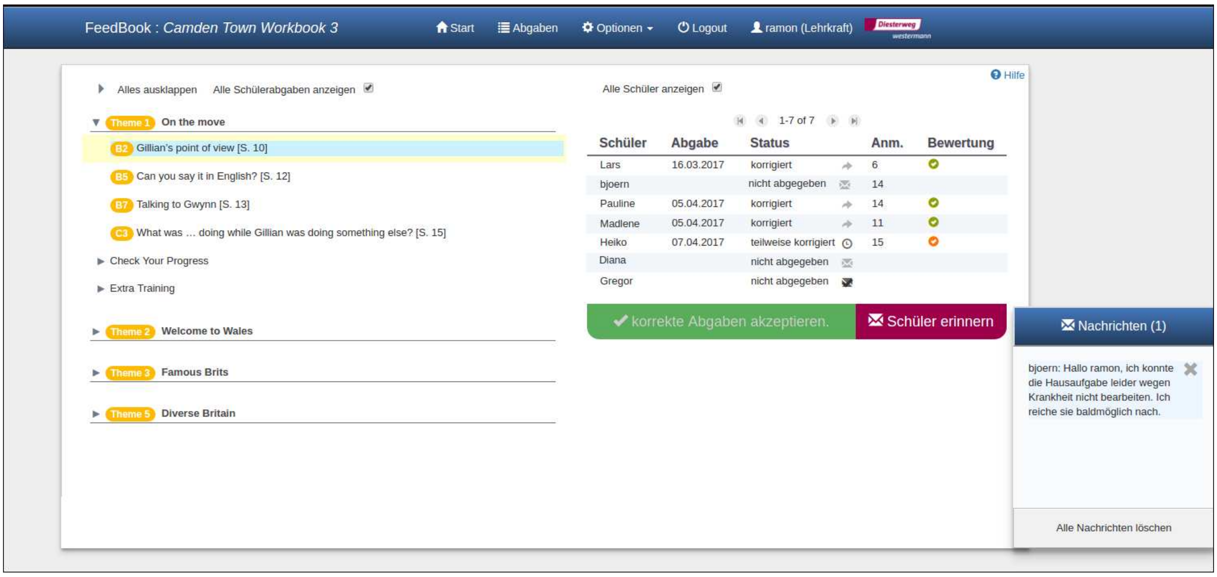
Speaking check: Can you say it in English?
Thema 5 CYP, Aufgabe 4
S.92

Grammar check: Questions for Leon
Thema 5 CYP, Aufgabe 6
S.93

Listening: Mrs Khan’s family
Thema 5 ET, Aufgabe 2
S.94

Mediation: Choosing a book
Thema 5 ET, Aufgabe 4
S.95

Figure 2: Student Lobby



Theme 1 On the move
- B2 Gillian’s point of view [S. 10]
- B5 Can you say it in English? [S. 12]
- B7 Talking to Gwynn [S. 13]
- C3 What was ... doing while Gillian was doing something else? [S. 15]
- Check Your Progress
- Extra Training

Theme 2 Welcome to Wales

Theme 3 Famous Brits

Theme 5 Diverse Britain

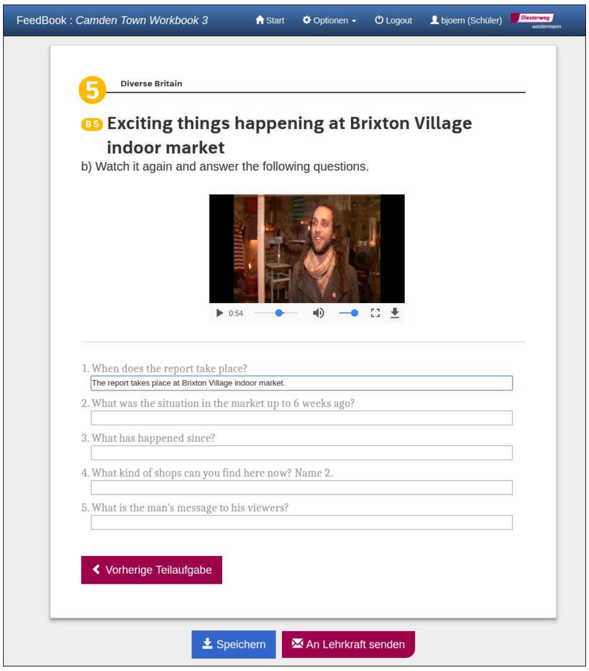
Schüler	Abgabe	Status	Anm.	Bewertung
Lars	16.03.2017	korrigiert	6	✓
bjoern		nicht abgegeben	14	
Pauline	05.04.2017	korrigiert	14	✓
Madlene	05.04.2017	korrigiert	11	✓
Heiko	07.04.2017	teilweise korrigiert	15	✓
Diana		nicht abgegeben		
Gregor		nicht abgegeben		

bjoern: Hallo ramon, ich konnte die Hausaufgabe leider wegen Krankheit nicht bearbeiten. Ich reiche sie baldmöglich nach.

Figure 3: Teacher Lobby

4.2 The Practice Interface for Students

The Practice Interface offers a fully functional web-based version of an exercise to students. For the print workbook exercise we saw in Figure 1, the web-based version is shown in Figure 4.



Diverse Britain

Exciting things happening at Brixton Village indoor market

b) Watch it again and answer the following questions.

1. When does the report take place?
The report takes place at Brixton Village indoor market.
2. What was the situation in the market up to 6 weeks ago?
3. What has happened since?
4. What kind of shops can you find here now? Name 2.
5. What is the man's message to his viewers?

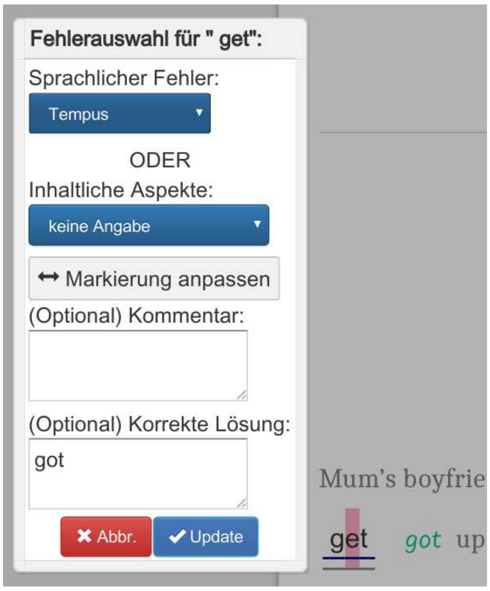
Figure 4: Student Practice Interface

The type of interactive input elements shown depends on the exercise type. If the workbook provides example answers, as often is the case for the first input field, these are visualized differently (unless the student changes them). If an exercise consists of multiple sub-tasks, the student can navigate freely between them. For all exercise types, the student has the functionality to save intermediate results and resume the exercise and all subtasks with any system input the student has provided for any number of trials. Once a student submits an exercise, the student is redirected to the lobby and the exercise is locked until the teacher has corrected it in the feedback interface.

4.3 The Feedback Interface for Teachers

The Feedback Interface is where the teacher has the opportunity to give feedback to student input on workbook activities. Starting out from the teacher lobby discussed in section 4.1, the teacher can view and select student submissions to particular activities. Once a student submission has been selected, the Feedback interface loads, rendering the student input in the context of the respective activity, in a view similar to the Practice Interface shown in Figure 4.

The teacher can then give both formative and summative feedback. The latter is given for the whole activity in the form of a star rating (1–5) and an optional global comment. Formative feedback can consist of specific annotations of the learner submission. Given its relevance to the overall goal of the FeedBook, we here describe it in more detail. In order to add an annotation, the teacher can select any sub-span of the student production, which opens the error annotation dialog shown in Figure 5.



Fehlerauswahl für " get":

Sprachlicher Fehler: Tempus

Inhaltliche Aspekte: keine Angabe

(Optional) Kommentar:

(Optional) Korrekte Lösung: got

Mum's boyfrie

get got up

Figure 5: The error annotation dialog

In this dialog, the teacher can specify four different characteristics: (i) the extent (span) of the error, (ii) the error type describing the nature of divergence from the norm, (iii) an example of a correct solution, and (iv) a free-text comment on this annotation. The error types we use are based on the categories provided by teachers in a pilot study. We grouped them into language form and content errors, as spelled out in Table 1.

Language form errors	Content errors
phrasing, agreement, determiner, preposition, grammar, spelling, pronoun, tense, clause structure, word choice, missing word, word order, punctuation	problematic understanding, missing information, wrong information, lack of understanding, extra information, alternate answer

Table 1: Error types in FeedBook

The annotations are saved in the database along with the student productions. They can be retrieved by several system components, in particular the Result Interface for students discussed in section 4.4.

4.3.1 Automatic Error Annotation Assistance

To assist teachers in providing feedback and reduce their workload, the FeedBook provides three distinct mechanisms: (i) a visual comparison with a pre-stored target response, (ii) a feedback memory storing the history of previously given feedback, and (iii) an automatic error annotation approach for some of the errors in the system. We discuss each of these features below.

Visual Highlighting of Answer Divergence In the current version of the system, each student answer is first string-matched against the pre-stored target answer; more complex matching approaches building on the research strands we compare in Ott et al. (2013) will be explored later in the project. If the match is positive, the student answer is assumed to be correct and no further processing needs to take place. This outcome is visualized to the teacher in the form of a green check mark next to the student answer.

If the answers differ from each other, a diff-like algorithm exemplified in Figure 6 is employed to find and highlight the parts of the student answer that differ from the target answer. This is done on two levels: First, for each target answer token, the nearest student answer token is identified via edit distance (Damerau, 1964). After this alignment step, the words thus matched are compared on a character level, and the differences in the student answer are highlighted. In order to avoid visualizing random meaningless similarities, highlighting is only performed if half the target answer material was identified in the student answer.

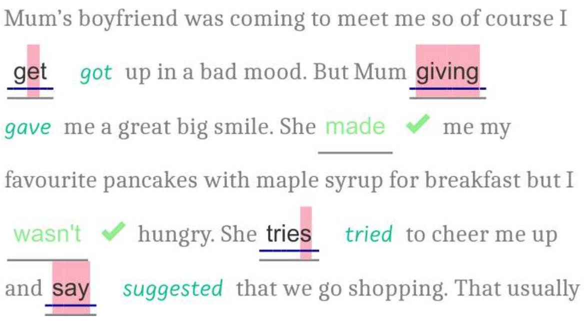

Mum's boyfriend was coming to meet me so of course I get got up in a bad mood. But Mum giving gave me a great big smile. She made ✓ me my favourite pancakes with maple syrup for breakfast but I wasn't ✓ hungry. She tries tried to cheer me up and say suggested that we go shopping. That usually

Figure 6: Difference highlighting example

Feedback Memory Depending on the activity type, many students will provide the same or very similar solutions to a given prompt. This is especially the case for more constrained types of activities. Ideally, the teacher should only have to provide feedback to such student answers once, and the system should remember previously given feedback to a given prompt and present it to the teacher at correction time. FeedBook achieves this by querying its database for feedback to the same prompts when the Feedback Interface is loaded, filling in any error annotations found for learner answers that were already annotated. This achieves both a reduction of the teacher's workload and a higher consistency in the teacher's error annotation.

Auto-correction In case there is no previous feedback from teachers, FeedBook provides automatic suggestions to the teacher based on NLP techniques. Eventually, this functionality will cover all activity and error types in the system, paving the way towards providing immediate feedback directly to the learner while they work on the exercise.

As a first step, we have implemented fine-grained feedback on the word level for grammatical errors in fill-in-the-blank activities. In order to perform error analysis reliably, the target answers are processed ahead of time in an NLP pipeline consisting of tokenization, sentence detection, POS tagging, lemmatization, morphological analysis, and dependency parsing. We use the Unstructured Information Management Architecture (UIMA, Ferrucci and Lally, 2004) along with the DKPro component repository (de Castilho and Gurevych, 2014), which provides access to many state-of-the-art NLP tools within an industry-strength processing framework supporting multiple layers of annotation. Table 2 lists the NLP components we use.

NLP task	**Component Used**
Tokenization, Sentence Detection, POS tagging	NLP Toolkit for JVM Languages (NLP4J[6])
Lemmatization	Morpha (Minnen et al., 2001)
Morphology	SFST with EMOR model (Schmid, 2005; Karp et al., 1992)
Dependency Parsing	MaltParser (Nivre et al., 2007)

Table 2: NLP components used in auto-correction

The automatically annotated linguistic information in the target answers is then manually post-corrected in order to provide a solid basis of reliable linguistic information for error diagnosis. We use WebAnno (Eckart de Castilho et al., 2016) for this purpose, which readily supports inspection and annotation of linguistic information produced by DKPro components.

When the teacher selects the Feedback Interface, the student answers are processed analogously to the target answers. On the basis of the annotated target answers, a sequence of rules compares student and target answer on different linguistic levels to determine the nature of the divergence to identify a non-word, a different POS, or a different inflection. This information on the divergence is then used as evidence for specific error annotations, such as *spelling*, *wrong word* or *tense* error. The first rule that applies stops the process, so once an error type has been identified, no more sophisticated comparisons are attempted.

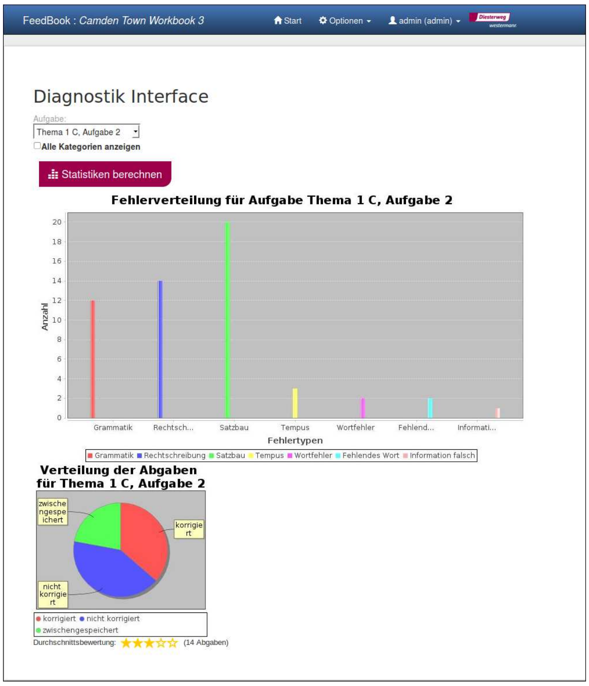
In case we cannot diagnose the nature of the error, a default error category is annotated, which is visually distinguished in the Feedback interface. The teacher can post-edit any automatically generated error annotations and change or remove them. In such cases of manual intervention, the system stores the now manually corrected annotations as instances for the Feedback Memory mechanism.

4.4 The Result Interface for Students

In the Result Interface, students see the selected activity with their submitted answers. Next to each student answer, the teacher's annotations are displayed if present. At the bottom, the system shows the global rating and comments by the teacher. In this interface, the student sees almost the same as the teacher in the Feedback Interface, the difference being that annotations are not editable and target answers are not shown (unless they were included by the teacher in the feedback).

4.5 The Diagnostics Interface

Teachers have access to a Diagnostics Interface, illustrated in Figure 7. After selecting a specific activity or an activity type, FeedBook offers three visualizations: A bar chart shows the quantitative distribution of annotations (errors) over the error categories introduced in section 4.3. This makes it possible to identify frequent problems in an exercise at one glance. Teachers can then use their class time more effectively to target these problems. For publishers this type of information would also be valuable for determining whether the errors made by students correlate with the pedagogical goal of the given activity or whether there are problems in the design of the exercise leading to unintended errors.



Aufgabe: Thema 1 C, Aufgabe 2

□ Alle Kategorien anzeigen

Durchschnittsbewertung: ★★★☆☆ (14 Abgaben)

Figure 7: Diagnostics Interface

The interface also contains a pie chart showing the distribution of the submission status for an exercise, i.e., what percentage of students worked on the exercise so far, how many exercises were submitted and await being corrected, and how many were already corrected. This chart thus serves as a progress indicator for a given exercise.

Finally, the Diagnostics Interface provides a visualization of the average global rating that the teacher gave to the submissions to a given exercise and the number of instances this is based on. The cases where students perform extremely low or high are especially interesting because they indicate a mismatch between the students' abilities and the activity demands.

4.6 Communication and Administration

For a web-based workbook, some communication and administration functionality is needed.

News System The FeedBook comes with a messaging service. For each activity submitted to a

[6] `https://emorynlp.github.io/nlp4j`

teacher, the student is shown a confirmation message in the news feed. When a teacher has corrected an activity, the student receives a clickable message, which will forward the student to the Result Interface for this exercise. Furthermore, teachers can write messages to each of their students, students can write to their teacher, and administrators can send messages to everyone. In the message, the name of the sender is shown.

The news feed widget always pops up in the lobby in case new (unread) messages exist, and the messages can be deleted by the recipient with a click on a cross displayed next to the text.

Account Management As the very first step in working with the FeedBook, teachers need to create accounts for their students. Student accounts can be created in a batch with the option to autogenerate passwords and to print the individual account information for distribution to the students in class. Once accounts exist, teachers have the possibility to reset their students' passwords in the account management interface in case a student forgets the credentials. Users can also change their own password. Teacher accounts are created by an admin user.

4.7 Meeting the Requirements of a Modern Web Application

Cross-Device Optimization To provide a convenient and motivating user experience encouraging users to interact with FeedBook, the system uses state-of-the-art web technology. A web application needs to provide an optimized display for all devices and platforms ranging from small mobile phones to wide monitors. For this, not only the size but also the positioning of elements needs to be adjusted. To build a fully scalable application, we use the Bootstrap[7] framework and its grid system for the display layer. We imitate a paper sheet via a stylized virtual paper as background that scales to all devices, in contrast to, for example, a PDF with a static width. Another challenge is to make the system work with mouse and keyboard input as well as with touch screens. For instance, in the Feedback Interface, where the teachers needs to select spans of the student answer, an additional selection slider menu is shown for touch events to support the same functionality available using a mouse.

Navigation The FeedBook features a permanently accessible navigation bar attached to the screen top. Among the functionality universal to all user groups it supports redirection to the lobby ("Start"), access to account management, and further options for some user groups. Teachers can navigate to the alternative lobby view ("Abgaben") discussed in section 4.1, and the options menu sports different drop-down menu items depending on the user group needs. The amount of information and the verbosity changes depending on the screen size, from a fully textual to an iconified mobile-friendly version for smaller devices.

5 Summary and Outlook

We presented the FeedBook, a web-based adaptation of a print workbook for English in 7th grade that aims at individually supporting students in completing both meaning- and form-based exercises. In the current stage, the system relies on the teacher to provide the feedback, but already considerably reduces the teacher workload through automatic visual highlighting, a feedback memory, and NLP-supported automatic error annotation suggestions. Moreover, the design of FeedBook as an online platform where students and teachers directly interact outside of the classroom enables teachers to flexibly assign activities and provide feedback.

The first version of FeedBook has been in use in pilot classrooms since October 2016 and feedback from teachers and students has been very supportive. We have put a premium on integrating feedback for new features and bug fixes in a timely fashion to ensure that the system continuously meets the demands of real-life classroom usage. For demonstration purposes, a separate instance of the system is available on the project website[8].

We are currently working on automating feedback for more types of student input as suggestions for the teachers. This will involve more complex NLP, especially for short answer tasks and content errors. In this context, we are exploring integration of a state-of-the-art short answer assessment system such as CoMiC (Meurers et al., 2011a) into the FeedBook.

In the next phase of the project, we will move the burden of giving feedback from the teacher to the system and redesign the FeedBook to provide

[7] http://getbootstrap.com

[8] http://feedbook.website

immediate, scaffolding feedback while the learner is completing the activity. The feedback data gathered from teachers up to this point will clearly be important for this development, and we will continue to give teachers the possibility to revise automatic feedback and provide manual feedback. In the third year of the project, we plan to conduct a randomized controlled field study integrating measures of the process and product of learning. We plan to compare the web-based workbook providing automated feedback with a web-based workbook transmitting only the teacher-provided feedback in order to evaluate the impact of immediate, scaffolding feedback on learning outcomes and motivation.

Acknowledgments

We gratefully acknowledge the support of this knowledge transfer project SFB833/T1 through the DFG. We would like to thank Christoph Golla and Laura Lanik, our project partners at the Westermann Gruppe, for the good cooperation. The project would also not be possible without the substantial contribution of the student research assistants: Verena Blaschke, Jekaterina Kaparina, Madeesh Kannan, Maxim Korniyenko, Johannes Krämer, Mai Mayeg, Samantha Tureski, Tobias Pütz, and Kuan Yu.

References

Luiz Amaral and Detmar Meurers. 2011. On using intelligent computer-assisted language learning in real-life foreign language teaching and learning. *ReCALL*, 23(1):4–24.

Luiz Amaral, Detmar Meurers, and Ramon Ziai. 2011. Analyzing learner language: Towards a flexible NLP architecture for intelligent language tutors. *Computer-Assisted Language Learning*, 24(1):1–16.

Fred J. Damerau. 1964. A technique for computer detection and correction of errors. *Communications of the ACM*, 7:171–176.

Richard Eckart de Castilho and Iryna Gurevych. 2014. A broad-coverage collection of portable NLP components for building shareable analysis pipelines. In *Proceedings of the Workshop on Open Infrastructures and Analysis Frameworks for HLT (OIAF4HLT) at COLING 2014*, pages 1–11, Dublin, Ireland. ACL and Dublin City University.

William DeSmedt. 1995. Herr Kommissar: An ICALL conversation simulator for intermediate German. In V. Melissa Holland, Jonathan Kaplan, and Michelle Sams, editors, *Intelligent Language Tutors: Theory Shaping Technology*, pages 153–174. Lawrence Erlbaum Associates Inc., New Jersey.

Richard Eckart de Castilho, Éva Mújdricza-Maydt, Muhie Seid Yimam, Silvana Hartmann, Iryna Gurevych, Anette Frank, and Chris Biemann. 2016. A web-based tool for the integrated annotation of semantic and syntactic structures. In *Proceedings of the Workshop on Language Technology Resources and Tools for Digital Humanities (LT4DH)*, pages 76–84.

David Ferrucci and Adam Lally. 2004. UIMA: An architectural approach to unstructured information processing in the corporate research environment. *Natural Language Engineering*, 10(3–4):327–348.

Trude Heift and Devlan Nicholson. 2001. Web delivery of adaptive and interactive language tutoring. *International Journal of Artificial Intelligence in Education*, 12(4):310–325.

Trude Heift and Mathias Schulze. 2007. *Errors and Intelligence in Computer-Assisted Language Learning: Parsers and Pedagogues*. Routledge.

Trude Heift. 2001. Error-specific and individualized feedback in a web-based language tutoring system: Do they read it? *ReCALL*, 13(2):129–142.

Trude Heift. 2003. Multiple learner errors and meaningful feedback: A challenge for ICALL systems. *CALICO Journal*, 20(3):533–548.

Trude Heift. 2004. Corrective feedback and learner uptake in call. *ReCALL*, 16(2):416–431.

Trude Heift. 2005. Inspectable learner reports for web-based language learning. *ReCALL*, 17(1):32–46.

Daniel Karp, Yves Schabes, Martin Zaidel, and Dania Egedi. 1992. A freely available wide coverage morphological analyzer for English. In *Proceedings of the 14th Conference on Computational Linguistics*, COLING '92, pages 950–955, Stroudsburg, PA.

Alison Mackey. 2006. Feedback, noticing and instructed second language learning. *Applied Linguistics*, 27(3):405–430.

Detmar Meurers and Markus Dickinson. 2017. Evidence and interpretation in language learning research: Opportunities for collaboration with computational linguistics. *Language Learning*, 67(2). `http://dx.doi.org/10.1111/lang.12233`.

Detmar Meurers, Ramon Ziai, Niels Ott, and Stacey Bailey. 2011a. Integrating parallel analysis modules to evaluate the meaning of answers to reading comprehension questions. *IJCEELL. Special Issue on Automatic Free-text Evaluation*, 21(4):355–369.

Detmar Meurers, Ramon Ziai, Niels Ott, and Janina Kopp. 2011b. Evaluating answers to reading comprehension questions in context: Results for German and the role of information structure. In *Proceedings of the TextInfer 2011 Workshop on Textual Entailment*, pages 1–9, Edinburgh.

Guido Minnen, John Carroll, and Darren Pearce. 2001. Applied morphological processing of English. *Natural Language Engineering*, 7(3):207–233.

Noriko Nagata. 1993. Intelligent computer feedback for second language instruction. *The Modern Language Journal*, 77(3):330–339.

Noriko Nagata. 1996. Computer vs. workbook instruction in second language acquistion. *CALICO Journal*, 14(1):53–75.

Noriko Nagata. 1997. The effectiveness of computer-assisted metalinguistic instruction: A case study in Japanese. *Foreign Language Annals*, 30(2):187–200.

Noriko Nagata. 2002. BANZAI: An application of natural language processing to web-based language learning. *CALICO Journal*, 19(3):583–599.

Noriko Nagata. 2009. Robo-Sensei's NLP-based error detection and feedback generation. *CALICO Journal*, 26(3):562–579.

Joakim Nivre, Jens Nilsson, Johan Hall, Atanas Chanev, Gülsen Eryigit, Sandra Kübler, Svetoslav Marinov, and Erwin Marsi. 2007. MaltParser: A language-independent system for data-driven dependency parsing. *Natural Language Engineering*, 13(1):1–41.

Niels Ott, Ramon Ziai, Michael Hahn, and Detmar Meurers. 2013. CoMeT: Integrating different levels of linguistic modeling for meaning assessment. In *Proceedings of the 7th International Workshop on Semantic Evaluation (SemEval)*, pages 608–616, Atlanta, GA. ACL.

Martí Quixal and Detmar Meurers. 2016. How can writing tasks be characterized in a way serving pedagogical goals and automatic analysis needs? *CALICO Journal*, 33:19–48.

Kent E. Sabo, Robert K. Atkinson, Angela L. Barrus, Stacey S. Joseph, and Ray S. Perez. 2013. Searching for the two sigma advantage: Evaluating algebra intelligent tutors. *Computers in Human Behavior*, 29(4):1833–1840.

Helmut Schmid. 2005. A programming language for finite state transducers. In *Proceedings of the 5th International Workshop on Finite State Methods in Natural Language Processing (FSMNLP)*, pages 308–309.